Amendments to the Claims:

Please amend claim 1 and cancel claims 10, 13-19, and 22-27 as follows.

1. (Currently amended) A computer-implemented method for assisting detection of critical memory leaks in a software program, the method comprising the steps of:

monitoring an amount of available memory for the software program during execution of the software program;

determining if the amount of available memory for the software program is less than a predetermined amount; ~~and~~

in response to such determination, storing a current stack walkback of each object currently referenced by the software program prior to the available memory dropping below an amount necessary to store the current stack walkback, wherein the current stack walkback assists in the detection of a critical memory leak during execution of the software program;

monitoring specified one or more analysis properties of the objects referenced by the software program, wherein the one or more specified analysis properties consists of at least one of ~~the~~ an object's age and an object's instance count;

determining if any analysis property of the objects being referenced following a garbage collection process exceeds a predetermined limit for such analysis property, wherein the predetermined limit for an object's age is an object age limit and the predetermined limit for an object's instance count is an object instance count growth value; ~~and~~

identifying any objects determined to have one or more analysis properties that exceeds property's predetermined limit; and

calculating an object's age by timing a current period starting when the respective object was instantiated.

2. (Canceled)

3. (Canceled)

4. (Peviously presented) The method according to claim 1, further comprising the step of calculating an object's instance count growth as magnitude of growth of an object instance count over a given time period.

5. (Previously presented) The method according to claim 1, wherein the step of monitoring comprises monitoring objects within a class designated for monitoring.

6. (Previously presented) The method according to claim 1, further comprising the step of performing the current stack walkbacks for the identified objects.

7. (Original) The method according to claim 6, further comprising the step of generating a statistics report including current stack walkbacks for the identified objects.

8. (Original) The method according to claim 7, further comprising the step of generating a web interface for user viewing of the statistics report at a computer display.

9. (Original) The method according to claim 1, wherein the software objects are Java objects.

Claims 10-27 (Cancelled).